日本オペレーションズ・リサーチ学会
2004年春季研究発表会

整数計画法を用いたポートフォリオ最適化

01102370 中央大学 今野浩 KONNO Hiroshi
02702020 中央大学 ＊山本零 YAMAMOTO Rei

1 はじめに

ポートフォリオ構築を行う際には、通常は様々な面倒な制約条件を単純化して解くのが一般的である。例えば、取引コストは非凸型の形状をしているため、取引コストの下でのポートフォリオ最適化問題は通常のアルゴリズムで解くことはできない。そこで多くの場合、取引コストを線形近似する方法が用いられている。しかし、取引金額が小さい場合には、近似誤差が大きくなる。

また、最小取引単位の存在もポートフォリオ構築問題を難しくする。連続変数を用いてポートフォリオ構築問題を解いた場合には、求めた解を近くの最小取引単位の倍数に丸めなければならない。しかし、取引金額が小さい場合には、最適なリターンーリスクの構造を崩してしまう可能性がある。

3つ目は銘柄数制約である。個人投資家の場合、管理できる銘柄数には限りがある。このため、組み込み銘柄を一定値以下とする問題を解くことが必要となる。

これらの条件は、すべて整数変数を導入することによって厳密に取り扱うことができる。

しかし、リスク指標として収益率の分散を使用すると、これらの問題は混合整数2次計画問題になる。実用規模の混合整数2次計画問題を解くことは容易ではない。

そこで、本研究ではリスク指標として収益率の絶対偏差を使用する。絶対偏差を使用すれば、制約条件を線形式に書き直すことができるので、問題は混合整数線形計画問題となり、標準的な方法を用いて解くことができる。

2 定式化

市場にはn種の資産があるものとし、構成するポートフォリオを$x = (x_1, x_2, \ldots, x_n)$とする。ポートフォリオ$x$の期待収益率と絶対偏差を、それぞれ$E[R(x)]$、$W[R(x)]$とすると

$$E[R(x)] = \sum_{j=1}^{n} r_j x_j \quad (1)$$

$$W[R(x)] = \sum_{t=1}^{T} \left| \sum_{j=1}^{n} (r_{jt} - r_j) x_j \right| / T \quad (2)$$

となる。ここでr_jは資産jの期待収益率、r_{jt}は資産jの第t期(もしくは第tシナリオの下で)の収益率である。

また、(2)式の絶対偏差は、補助変数ϕ_t、ψ_tを導入することによって、以下のように書き直すことができる。

$$\begin{aligned}
&W[R(x)] = \sum_{t=1}^{T} (\phi_t + \psi_t)/T \\
&\phi_t - \psi_t = \sum_{j=1}^{n} (r_{jt} - r_j) x_j,\ t = 1, 2, \ldots, T \\
&\phi_t \geq 0,\ \psi_t \geq 0,\ t = 1, 2, \ldots, T \\
&\phi_t \psi_t = 0,\ t = 1, 2, \ldots, T
\end{aligned}$$

ここで相補性条件$\phi_t \psi_t = 0$は、後の定式化の際(次ページの式(3))では取り除くことができる。

(a) 取引コスト関数

通常、取引コスト関数は区分線形凹型もしくは階段型の形状をしている(図1)。

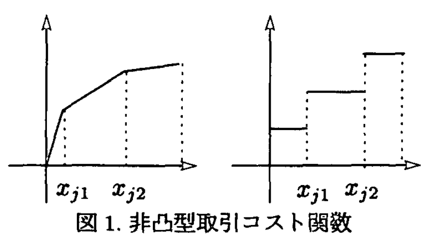

図1. 非凸型取引コスト関数

ここでは紙面を節約するため、階段型のコスト関数の定式化のみを示す。0-1変数$y_{jl}, l = 1, 2, \ldots, k_j$を導入すると、コスト関数$c(x_j)$は以下のように表

現することができる。

$$c(x_j) = \sum_{l=1}^{k_j}(c_{jl} - c_{jl-1})y_{jl}$$
$$\frac{x_j - x_{jl}}{u_j} \le y_{jl} \le 1 + \frac{x_j - x_{jl}}{u_j}$$
$$y_{jl} = 0 \text{ または } 1,\ l = 1, 2, \ldots, k_j.$$

ここで u_j は資産 j への投資上限である。

(b) 最小取引単位制約

資産 j の最小取引単位を γ_j、非負の整数変数を z_j としたときに、最小取引単位制約は以下のように表現される。

$$x_j = \gamma_j z_j,\ j = 1, 2, \ldots, n.$$

(c) 銘柄数制約

資産 j を購入する場合には必ず $y_{j1} = 1$ となる。したがって、p 種の銘柄でポートフォリオを構築するという条件は、以下のように表すことができる。

$$\sum_{j=1}^{n} y_{j1} = p.$$

これより、絶対偏差を一定値 w 以下としたときに、ネットリターン (リターン − コスト) を最大化するポートフォリオ構築問題は、以下のように定式化することができる。

$$\begin{array}{ll}
\text{最大化} & \sum_{j=1}^{n} r_j x_j - \sum_{j=1}^{n}\sum_{l=1}^{k}(c_{jl} - c_{jl-1})y_{jl} \\
\text{条件} & \sum_{t=1}^{T}(\phi_t + \psi_t)/T \le w \\
& \phi_t - \psi_t = \sum_{j=1}^{n}(r_{jt} - r_j)x_j \\
& \frac{x_j - x_{jl}}{u_j} \le y_{jl} \le 1 + \frac{x_j - x_{jl}}{u_j} \\
& \sum_{j=1}^{n} y_{j1} = p \\
& \sum_{j=1}^{n} x_j = 1 \\
& y_{jl} = 0 \text{ または } 1,\ 1,\ l = 1, 2, \ldots, k; \\
& \qquad j = 1, 2, \ldots, n \\
& \phi_t \ge 0,\ \psi_t \ge 0,\ t = 1, 2, \ldots, T \\
& x_j = \gamma_j z_j,\ j = 1, 2, \ldots, n \\
& 0 \le x_j \le u_j,\ j = 1, 2, \ldots, n.
\end{array} \tag{3}$$

3 計算機実験

次に問題 (3) を CPLEX7.1 を用いて解いた結果について説明する。表 1 には、$w = 0.035$、$u_j = 0.02$ とおき、ユニバースの銘柄数 n とポートフォリオに入れる銘柄数 p を変化させたときの結果を示した。なお計算機環境は、CPU が 1.67GHz、メモリが 512Mbyte である。

$p \setminus n$	200	400	600	800
55	33	52	705	1754
70	13	79	124	996
100	102	158	99	639
130	61	265	50	7908
160	43	45	89	2639
190	21	90	216	5926

表 1. 計算時間 (秒)

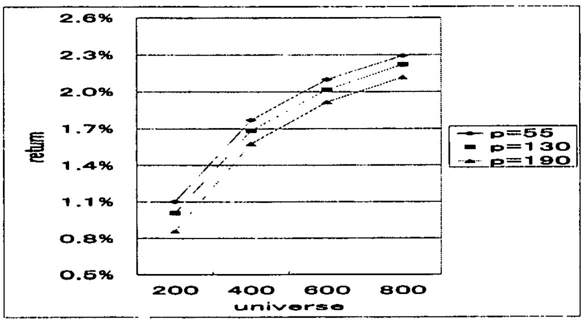

図 2. ネットリターン

4 考察と結果

これらの結果より、様々な実務的制約のもとでのポートフォリオ最適化問題を、実用的な時間で解くことができることがわかった。また、ポートフォリオに組み込む銘柄数は、多いほどよいというものではなく、比較的少数銘柄で最適なリターン-リスク構造を作ることができることがわかった。

当日の発表では、詳細な計算機実験の結果と、本モデルをインデックス・トラッキング・モデルに拡張した場合の実験結果についても報告する。

参考文献

今野 浩「理財工学 II」, 日科技連出版,1998.